Journal of Secure Communication and System (2017)

ORIGINAL RESEARCH ARTICLE

Research on Secure Communication Based on QQ Chat Platform

Yi Cao,Hao Tang,Jiangang Zhou

Software Research and Development Center, College of Computer Science, Ningde University, Fujian, China

ABSTRACT

QQ is China's most widely used instant messaging tool, its security for user security and network security has an important impact. Once a QQ account is hacked, the harm and the impact is terrible, the user loses a lot of friend's information and contact information in a short time and is difficult to completely recover. This article will analyze the security of QQ's landing agreement, analyze its security, and point out the security vulnerabilities which exist, and explain the possible attacks that QQ may suffer, and make suggestions for improvement and ways to enhance the security of QQ communication. In addition, this article also conducts a corresponding study of QQ hacking, and puts forward precautionary approaches.

KEYWORDS: QQ; login agreement; security vulnerability; improvement advice

Citation: Cao Y, Tang H, Zhou JG, et al. Research on Secure Communication Based on QQ Chat Platform, Journal of Secure Communication and System (2017); 1(1): 1–9.

Correspondence to: Jiangang Zhou, Software Research and Development Center, College of Computer Science, Ningde University, Fujian, China, zjgcoom@sina.com.cn.

1. QQ Login Protocol

QQ login protocol is Tencent's own development of binary data based on the application layer network protocol, QQ login protocol uses a fixed format, where there is only a little difference between the message format sent between the client and the serve. The symmetric encryption algorithm used in the protocol is TEA (Tiny Encryption Algorithm), which is a random encryption function. It is called QQTEA. The hash function is MD5. QQ login is divided into UDP and TCP login, also supports proxy login. In most cases UDP login is used. The number of UDP login port servers are 8000, and the local port is generally selected starting from 4000, if the port has been occupied, then 1 is added and then tested, until it finds a port that is not occupied.

TCP login server generally has 443 ports, the local port selection and is similar to the UDP way. Login mode can be set freely in the login settings, but no matter which way is used to log in, the login process and packet format are the same.

QQ login is mainly divided into the following six steps:

(1) Touch packet

This packet is the first packet sent during the QQ client login, its role is to test whether the remote server can respond.

(2) To obtain a verification code

As part of the QQ login ID may be saved off-site , or because the QQ login ID stolen to send a lot of spam, or used on-hook software or was hang up by hang up sites, Tencent servers on detecting such abnormality while using QQ will require the user to input their verification code

(3) Password verification

The role of this packet is the local QQ password sent to the server for verification.

(4) Verification Token 1

This packet is mainly used to verify the first few packets of Token data, if verified, the server will return to the login time and IP address and other related information.

(5) Verification Token 2

(6) To obtain the session key

This packet is the last packet sent in the QQ login process, mainly for the QQ client to return the session key which is generated by the server.

Each step login instruction is: 0x0825, 0x0826, 0x0826, 0x0826, 0x0826, and 0x0828.

Related Symbol Description:

RKey: A key used to represent a randomly generated 16-byte length.

En (Key, P): indicates the key used to encrypt the plaintext P, here meaning the QQTEA encryption.

Datai: Data with no clear or specific meaning. (I takes a positive integer)

Keyi:Represents a 16-byte length key. (I = 0, 1, 2, 3, 4)

MD5: it is a hash function.

M2P: Md5 (Md5 (QQPassword)).

QQPassword: QQ password

SessionKey: Session key.

2. QQ login protocol security

2.1. Introduction to TEA

The TEA algorithm was invented by David Wheeler and Roger Needham of the Cambridge University Computer Laboratory in 1994. TEA is an abbreviation for Tiny Encryption Algorithm. It is characterized by its fast encryption speed, high speed andefficiency, but poor anti-differential attack capability. The TEA encryption algorithm is a block cipher algorithm with 64 bytes (8 bytes) of plaintext block and 128 bits in length (16 bytes). The number of iterations of the TEA encryption algorithm can be changed. The number of iterations is 32 rounds and the two TEA Feistel cycles are counted as 1 round.

TEA algorithm is widely used in QQ data encryption. QQ uses 16 rounds of TEA algorithm encryption. It currently uses 16 rounds of encryption instead of the standard 32 rounds of encryption in order to reduce the pressure on the verification server. Before data encryption QQsutilizes commonly used filling and interleaving technology to reduce the relevance of encrypted data, increase the cracker's difficulty.

TEA algorithm code is as follows:

```
void qq_encipher(unsigned long * const plain, const unsigned long * const key, unsigned long * const crypt)
// The parameter is an 8-byte plaintext input and a 16-byte key that outputs 8 bytes of ciphertext
{nsigned long left = plain [0], right = plain [1],
a = key [0], b = key[1],
c = key [2], d = key[3],
n = 32, sum = 0,
delta = 0x9E3779b9;
// plaintext input is divided into two parts, the key is divided into four parts into the register, n represents the number of encrypted rounds recommended 32. Delta is a constant.
while (n--> 0) {
sum += delta;
left += ((right << 4) + a) ^ (right + sum) ^ ((right >> 5) + b);
right += ((left << 4) + c) ^ (left + sum) ^ ((left >> 5) + d);}
```

```
crypt [0] = left;
crypt [1] = right;
}
void decrypt (unsigned long * v, unsigned long * k) {// decryption process
unsigned long y=v [0], z=v[1], sum=0xC6EF3720, i;
unsigned long delta=0x9e3779b9;
// delta gold split rate. Its effect is to make each round of encryption different. Initialize to 0x9e3779b9
unsigned long a=k[0], b=k[1], c=k [2], d=k [3];
for (i = 0; i <32; i ++) {// loop entry
z -= ((y << 4) + c) ^ (y + sum) ^ ((y >> 5) + d);
y -= ((z<<4) + a) ^ (z + sum) ^ ((z>>5) + b);
sum -= delta; / * end loop * /}
 v[0] = y;
v[1] = z;}
```

2.2. QQTEA algorithm

In the first chapter of this paper, the symmetric encryption algorithm used in the protocol is TEA (Tiny Encryption Algorithm), which is a random encryption algorithm, which is called QQTEA. So here will be mainly introducing filling, interweaving and feedback, if you are interested in the encryption process of the TEA algorithm, can refer to the relevant literature [1].

(1) Fill algorithm

In order to apply the TEA algorithm, it is necessary to make the number of bytes in plaintext a multiple of 8. If the length of the plaintext itself is not a multiple of 8, then it is also filled so that it becomes a multiple of 8. In bytes, so that N = the original string +10 + fill the number of bytes n, then N should be a multiple of 8.

The first byte is: (random () \u0026 0xf8) | n, followed by (n + 2) bytes random () \u0026 0xff, followed by the original data, and finally filled with 7 bytes 0x00. Because of the use of different random numbers, the result of the filling makes the cipher text results different even for the same plaintext.

1Byte fill length (n) +2 plaintext length 7Byte

Random number fill length random number plaintext 0x00

Random () \u0026 0xf8 n random () \u0026 0xf8

3Byte

Fill length (n) = 8 - ((plaintext length +2)% 8)

(2) Interweaving algorithm

The message is divided into multiple encryption units, each encryption unit is 8 bytes, TEA is used for encryption, and the encrypted unit and the next encryption unit either are similarly encrypted or after the operation are used as plaintext to be encrypted.

(3) Feedback encryption

Because TEA is a packet encryption algorithm, it is necessary to block the plaintext. Plain [i] represents the i-th packet of the plaintext, crypted [i] represents the i-th packet of the cipher text, all packets are encrypted using the same key and are assigned as “key”. The specific feedback encryption process is shown in the following figure:

The following are the same as the ‘

When i = 1, crypted [i] = E (key, plain [i]);

When i> 1, crypted [i] = plain [i-1] xor E (key, plain [i] xor crypted [i-1]).

In the reference text [1] it is mentioned that when verifying that a key is correct, only the last 16Byte of the text and cipher text (that is, two groups) on the line. However, according to the formula given above, we will find that for the first group, in fact, feedback processing was not carried out, therefore, only the first 8 bytes (that is, a packet) is used for decryption, and through comparison can also determine the validity of the key.

2.3. PseudoRandom Number Generator

In the QQ2012 client programa pseudo-random number generator (PRNC)was used, and the following briefly describes the pseudo-random number and its security.

(1) Pseudo-random number

The random number (or random event) in the real sense is randomly generated in the course of a given process according to the probability of distribution in the course of the experiment. The result is unpredictable and invisible. Conversely the computer's random function is simulated according to a certain algorithm, the result is determined, and is visible. So the random number generated by the computer random function is not random, and hence is a pseudo-random number.

(2) Pseudo-random number generator

The pseudo-random number generator (PRNG) is a method of generating pseudo-random numbers. In a lot of encryption algorithms and many security protocols are involved in the generation of random numbers, it can be seen, a safe pseudo-random number generator for cryptography is essential. Therefore, it is necessary to study the security of pseudo-random numbers, but because of the determination of the computer, the security of pseudo-random numbers is often referred to as unpredictable in polynomial time. It should be noted that, the pseudo-random number generator used in the QQ2012 client program is a linear congruence generator LCG . The QQ2012 client uses the LCG from the Microsoft Visual / Quick C / C ++ rand () function. The specific calculation method is as follows: Xn = (Xn-1 * A + B) mod M

Where Xn is the nth number of the sequence, Xn-1 is the number n-1 of the sequence, and A, B, and M are constants (which are usually prime numbers). When B = 0, it is called as a multiplication method. When related to a concept called seed, it will be replaced by X0 as above, and then each time the call rand () function will be used to generate a random value to generate a new random value. It can be seen that the the rand () function is actually a recursive sequence, and all values are derived from the original seed. So when the initial seed is the samethe same sequence will be obtained.

Rand () function Prototype:

```
Int __cdecl rand (void) {
Return (((holdrand = holdrand * 214013L + 2531011L) >> 16) \u0026 0x7fff);
}
```

If you want to use the linear congruence generator LCG to generate a random key, then there are two ways: the first is to produce two bytes each time, the other is to produce a byte each time. As a generic function routine, the second method should be used, and the specific function routines are as follows:

```
Int fillrandom (char * buffer, const int size) {
    Int i;
   For (i = 0; i <size; i ++)
{Buffer [i] = rand () \u0026 0xff;}}
```

It can be learned from the above that when the appropriate A and B are selected, the resulting sequence period can reach M. For 32-bit programs, this cycle is still safe. However, if the key is generated according to the above routine, the period is greatly shortened because only the 8 to 15 bits (the least significant bit is 0) are used. This produces a complete cycle with a length of 2 ^ 24. It is absolutely possible to retrieve this sequence within the allowed time range, which provides an opportunity for offline dictionary attacks.

3. QQ login protocol loopholes and improvements

3.1. Vulnerability 1 - Pseudo-Random Number Generator Attack

In the above, the pseudo-random number and the pseudo-random number generator used in the QQ2012 client program have been briefly introduced, and the security is analyzed. It can be seen, because the pseudo-random number

generator has security vulnerabilities, we can effectively predict the generated random key. From the introduction to the QQ login protocol in Chapter 1 of this article, we can get {Key1, Key2} once we predict Key0; get the Key2, and then know the {Key3, Key4}; SessionKey. So, we can understand this: the Key0 attack, is in fact, for the SessionKey. Therefore, the direct study of the important SessionKey and the attack on it for QQ login protocol is sufficient. The following focuses on the attack on the SessionKey (the relevant data can refer to Chapter 1 QQ login protocol flow chart):

A. Start capturing pcap_loop (UDP packets)

B. To determine whether the success of the capture packet, it is c, or re-capture

C. Determine whether the packet meets the conditions:

Dport = 8000 and contains RKey, E (RKey, En (RKey, En (M2P, Key0))); if satisfied find Rkey in the Rand.bin and offset ,save offset, and then advance to step b; if not satisfied proceed to D.

D. Determine whether the packet meets the conditions:

sport=8000 and contain En(Key0, {Key1, Key2}), if fulfilled offset from Rand.bin and try every possible key. Once founduse Key0 to decrypt En(Key0, {Key1, Key2}), Get and save Key2, and then proceed to step b; if not satisfied proceed to E.

E. Determine if the packet meets the criteria:

Sport=8000 and whether contain En(Key2,{Key3,Key4}0; If fulfilled used Key2 to decrypt En(Key2, {Key3, Key4} to obtain and save Key3, and then perform step b; if not satisfied, then proceed to f.

F. Determine whether the packet meets the conditions:

Sport=8000 and contain En(Key3, SessionKey); if satisfied use Key3 to decrypt En (Key3, SessionKey) to obtain and save Key3; if not satisfied, then proceed to B.

Related notes:

dport: Specifies the destination port of the UDP packet

sport: Specifies the source port of the UDP packet

Rand.bin: Use the method mentioned in the previous section to generate a complete cycle of the random number of the data file.

Find the offset of the RKey: You can use the violent method Rand.bin size is only 16MByte. If this cycle is relatively large, through a reasonable organization of data structures and pre-calculation, you can find the corresponding time in the faster offset.

Test each possible key from offset: Passing multiple tests to find Key0 is in offset-16 offset position. Complex client login operations may change this offset, so the search is required, and the offset-16 offset position is correct in most cases.

Include: only according to the UDP packet of the first four and five bytes to judge.

It should also be noted that this attack is not applicable for every circumstance, at least in the ‘login protection’ case it is invalid. This is because the Key0 under ‘Logon Protection’ should be at least one MD5 checksum associated with the login IP.

3.2. Vulnerability 2 - Offline Dictionary Attack

The so-called dictionary attack, refers to the collection of good passwords may contain the string, and then through a variety of combinations, is tested one by one. In fact, it is akin to just guessing the password, but using a computer to complete. In the reference [1] a QQ2008 offline dictionary attack method was proposed, and in reference [2] also briefly mentioned was a QQ2010 offline dictionary attack, in fact, these methodsapplies for QQ2012 and shall not be repeated in this text.

Referring to Chapter 2 which is on the analysis of QQTEA algorithm, it can be learned that in order to verify a password, in addition to testing the last 16Bytes, we can first take the first 8Byte for a decryption test. As a result, the first packet is not fed back, and the lowest three bits of the first byte contain known information (the number of bytes that are randomly filled). In fact, if any two consecutive packets in the plaintext are known, then they can all be used for decryption testing. The following will explain the test key verification procedures (for the relevant data refer to Chapter 2 QQTEA algorithm):

A. Enter the test key , the first 8 bytes of the cipher text crypted

B. Use the test key to decrypt plaintext plain = D (key, crypted)

C. To determine whether the conditions are met: plain [0] \u0026 0x7 is equal to 1; if the conditions are met, then proceed to D; otherwise, return the message 'wrong key'.

The reason for this judgment: For the correct authentication, the size of the packet returned by the server is always 285 bytes, according to the fill rules, the minimum 8 bits of the first byte to be filled should be equal to (8- (285 + 2)% 8) = 1

D. Determine whether the conditions are met: plain [4,5,6,7] is equal to 0x01,0x19,0x00,0x00; if the conditions are met, then proceed to E; otherwise, return the message 'wrong key'.

The reason for this judgment: For the correct authentication, the first four bytes returned by the server are fixed.

E. Determine if the condition is satisfied: the last 7 bytes of plain are 0x00; if the condition is satisfied, the message 'found the correct key' is returned; otherwise, the message 'wrong key key' is returned.

The reason for this judgment: In fact, the two judgments of step C and step D are sufficient to determine the correctness of the test key, but this judgment is added for a more rigorous process.

D (key, crypted): Indicates that the key is used for the cipher text.

3.3. QQ Login Protocol Improvement Suggestions

The improvements presented in this chapter are mainly for the two attacks mentioned above.

1. Defense against the pseudo-random number generator attack method

According to the above description, if you want to protect the pseudo-random number generator attack, just modify the random number to generate the key (here involves two points: sequence cycle and K0), then the specific implementation is as follows:

(1) cycle: the use of a longer cycle of random number generator.

(2) K0: On K0, it is mainly to increase the difficulty of being predicted. From the Chapter 1 QQ login protocol flow chart we can see, K0 generated random number parameters. In fact, in addition to the parameters of a random number, together with MD5 ({QQNumber, QQPassword, Time}) an exclusive-OR operation can be conducted, so that the attacker cannot easily predict a random number to get K0 information.

2. Defense against offline dictionary attacks

Assuming that the attacker uses an offline dictionary attack, then there is no more information on the difference between the attacker and the legitimate user. First, we try to use a secure key exchange protocol to resist offline dictionary attacks, but the key exchange protocol currently used is at least Diffie-Hellman Key Exchange / Agreement Algorithm. The cleverness of this protocol is that both parties who need secure communication can use this method to determine the symmetric key and then use that key for encryption and decryption. But this type of agreement requires a certain amount of storage and calculation, and not suitable for some QQ terminals. In Chapter 3, dictionary attacks are referred to as collecting a passwords that may contain the string, and then through a combination of various ways, one by one tested, in order to obtain the correct key. Then we can resist the attack by increasing the complexity of dictionary attacks:

(1) Increase the number of TEA rounds(since the number of iterations of the TEA encryption algorithm can be changed, the number of iterations is 32 rounds, and the QQ is encrypted with 16 rounds of TEA)

(2) modify the end of the text of the additional 7 bytes of the contents of the fill for the random number

4. QQ hacking and prevention

4.1. QQ hacking

QQ as the most widely used instant messaging tool in the country, in addition to the large number of users, the number of malicious programs for it is also numerous. Most of these malicious programs steal QQ login ID and password, put the account up for sale, or further the implementation of cybercrime or fraud. Currently a common and effective way to hack is to use 'QQ hacking Trojans'. Daohao Trojan is a Trojan horse with the ability to invade the computer and steal QQ passwords when browsing malicious websites or when opening files containing the virus,

When the Trojan is loaded, it will regularly check whether the QQ is running. If it is running, it displays a fake login window, prompting the user to enter the account number and password to log in again. If the user accidentally follows the prompt operation, the Trojan will record the user's QQ account and password, and send it to the hacker's pre-specified address.

QQ Daohao Trojan's general operation mode is: to monitor the operation of the user login window → through a remote port save the operating record to a pre-set location → the producer receives, the password and account to steal → modify the password, loot all valuable virtual items.

There is also a practice, that is, when the user opens the login window, the virus attempts at hundreds of thousands or even millions of times per second speed to obtain the password, until after it success, then sends the correct account password to the producer.

The most recent discovery was when a laboratory recently intercepted a Trrojan with the filename Trojan-PSW. Win32.QQPass.binz that will steal the user's QQ information and send it to remote hackers. Unlike the common Trojans, the Trojan is a DLL file. The DLL file needs an .exe file to load to run. The Trojan writers do not write their own DLL loader, but modified some of the system files such as msdtc.exe to load. It is possible to avoid some of the malware protection software's active inspection, so this method of Daohao's success rate greatly increased.

When a QQ account is stolen, the harm and impact is terrible, the user loses a lot of friend's information and contact information in a short time which is difficult to completely recover. The user's privacy is leaked, but may also cause economic losses, as well as their investment of a lot of energy to improve the account is lost.

4.2. Commonly used Daohao Methods and Software

Method 1: Posing as QQ user's relatives, friends, classmates, etc., to QQ users to send information. For example: XX I went to XX website to see a very interesting article, please have a look. And after leaving a URL, when you click to enter the site on entry when you are prompted to enter the QQ number and login password, it is difficult to think this is Daohao people left the hacking window, Daohao people from the background can see the login password. There is also a method which shows a popup that QQ login has failed please re-login, and hence the person is hacked.

Method 2: Trojans have a wide variety, and they are installed in the user's computer like a bug, and constantly monitor the user's computer every move, who then unconsciously put what the hackers are interested in: the account password information which are then sent to hackers For example, a QQ thief will installed in the user's computer a key logger to intercept the users input when they enter their QQ account password information which is then sent to the hacker's server.

Method 3: There are some Trojans that disguise themselves as a key module in the computer operating system. And then deceive a lot of software to make a lot of normal software into Daohao Trojan accomplices. Hackers usually use Daohao Trojans dressed as msimg32.dll (see pictures related to important files) and other software essential modules, and then are hidden in the software installation directory such as QQ, when the use of these processes are vital with the running of msimg32.dllin accordance with the rules of the operating system these puppet software become a Trojan horse (fake msimg32.dll) accomplice.

Method 4: There is a very popular way to hack QQ information through the QQ chat when a single user account stolen, all his friends, QQ group will receive the phishing message, the content format is 'XX needs friends tohelp + phiishing website '. If someone click on the phishing site to open fake QQ space, the background is usually a beautiful photo album thumbnail, an in an eye-catching prompt users are prompted' to log in to access QQ space, with the purpose to cheat the user's QQ account and password and then ensnaring more of their friends.

Hackers commonly used hacking software:

1, QQ robot

QQ robot is a online decryption tool that can save multiple account passwords, if the user's QQ password accidentally lost, you can use the software to retrieve the account password.

2, QQ Simple Theft

QQ Simple Theft is a classic hacking software that using the insertion technology, it itself does not produce the process, it is difficult to be found, it will automatically generate a Trojan horse, and as long as the generated Trojan is sent to the target user, and trickeries able to trick the user to run the Trojan file, it has successfully achieved the purpose of the invasion.

3, Mini QQ Password Interceptor

Mini QQ password interceptor is a highly efficient QQ password interception software, it will open the user in the QQ program 'Registration Wizard' and uses this opportunity to easily steal the QQ ID and password, after which the program will automatically send the number and password to the specified mailbox.

4, Good Friends Good Pirates

Good friend is a very useful remote Daohao software, which can be used on friends who are online to steal their ID and password.. The software uses graphics practically and sends passwords directly through the QQ in addition to having encryption with information transmission function.

5, Ala QQ thief

Ala QQ thief is a QQ password theft tool, which is highly disruptive and hidden, and the tool steals passwords through the use of transfer of password information and camouflage icons, etc.

6, QQExplorer

QQExplorer is a more commonly used online crack QQ password tool, is powerful, and easy to set up.

7, Stolen Q Black Xia

Pirates of the Q Black Xia is a special QQ password theft tool, while the program is running, they separateinto three files which safeguard each other and with , the termination of one it will automatically recover, thereby increasing the rate of theft of the user.

8, The Edge of Ice

Ice edge is a very strong camouflaged QQ Trojan horse, which can be customized to set to displayerror message when QQ isd hacked, so that the program and its hacking is more well hidden. You can also set up so that while the Trojan is running, it automatically opens a file text..

9, QQ cracker messenger

QQ cracker messenger is a local crack QQ password hacker tool. The user can choose the dictionary violent crack for local QQ password theft.

4.3. ways to Prevent Hacking.

1, Apply for password protection for your own QQ account.

2, Be careful of pages linked to you, and all prompts to enter the QQ number and login password. Do not enter, unless it is someone you know personally or is a secure website.

3, To prevent the Trojan invasion of computers, timely install anti-virus software.

4, Bind QQ security center.

5, Bind the security card (not recommended) and mobile phone.

6, The use of more complex and easy to remember passwords (recommended to regularly update the password), and try to avoid using the same QQ password in other sites.

7, When a stranger sends to the unknown software, do not open. If you need to open, please use a virtual image to open.

8, If you enter a web page and QQ suddenly closes or logs out we must pay attention and do not relog in. Run antivirus and after confirming you are safe, only log in.

9, Improve other aspects of security awareness (with particular attention to prevent the page containing personal information).

5. Conclusion

Based on previous research and analysis, this paper discusses the security of communication of QQ, mainly through the study of the security of QQ's login protocol, and analyzes its loopholes, and gives the relevant improvement suggestions, and also introduced the current common QQ Daohao method and Daohao software and has given several precautionary approaches. In the whole paper design process, there have been some problems, but through the consultation teachers and students and inquiries related information successfully resolved. Through this paper design, I have a deeper understanding on cryptography and network security, and I also deeply appreciate that to do a complete

a task, the need for a systematic way of thinking and methods, in the face of the need to for problems solving, to be patient, but also be good at using the existing resources to enrich ourselves.

References

1. Yi Zongxiang. QQ login protocol security analysis and improvement research, 1671-1122 (2011) 06-0085-03
2. Yu Kai, Zhang Yi, Wang Yongjun. QQ login protocol security research and analysis [J]. Information Network Security, 2008, (11): 55-57.
3. Tencent QQ security channel, http: //im.qq.com/safe/index.shtml
4. QQ working principle and encryption, hacking. 2011-07-06 01: 36QQ communication protocol introduced
5. Wu Shizhong translation, Application of cryptography (protocol algorithm and C language program), Machinery Industry Press, January 2000
6. Qiu Zhongpan translation, Cryptography and network security, Tsinghua University Press, September 2005
7. Yang Xiaoyuan, Computer cryptography, Xi'an Jiaotong University Press, March 2007
8. QQ agreement analysis - login
9. Http://blog.sina.com.cn/s/blog_4b27f27e01000816.html
10. QQ2013 agreement shallow analysis http://my.oschina.net/fsxchen/blog/129356